

IN THE CLAIMS:

Please cancel without prejudice Claims 1-57. Please add the newly drafted Claims 58-92.

1-57. (Cancelled)

58. (New) An application execution apparatus including a plurality of devices, comprising:

a storing unit for storing a plurality of applications;

a kernel unit for controlling execution of the plurality of applications in the storing unit; and

a plurality of library units, corresponding one-to-one to the plurality of devices, each for managing a corresponding device by executing a device control program,

wherein each library unit includes:

resource providing means for, when an application being executed requests provision of a device corresponding to the library unit, providing the device to the application as a resource, and registering a correspondence of the resource and the application provided with the resource into a table;

requesting means for, if the resource providing means provides the resource to the application, requesting the kernel unit to notify the library unit of completion of the application when the application is completed; and

collecting means for, when the kernel unit notifies the library unit of the completion of the application in response to the request from the requesting means, specifying the resource corresponding to the application based on the table, and collecting the specified resource, and

the kernel unit includes:

application executing means for reading the plurality of applications in the storing unit and executing the plurality of applications; and

notifying means for, when the application being executed is completed, notifying the library unit which has requested the notification, of the completion of the application.

59. (New) The application execution apparatus of Claim 58,

wherein the requesting means requests the notification, by registering a callback function into the notifying means, and

the notifying means notifies the library unit of the completion of the application by calling and executing the registered callback function, when the application is completed.

60. (New) The application execution apparatus of Claim 58,

wherein the notifying means generates an application information instance storing application information for identifying each application which is in execution,

the resource providing means generates, when the application being executed requests the provision of the device, a resource collection instance storing the table and a method of retrieving information from the table,

the requesting means requests the notification, by registering the resource collection instance into an application information instance storing application information for identifying the application provided with the device as the resource, and

the notifying means notifies the library unit of the completion of the application by calling the method in the resource collection instance registered in the application information instance by the requesting means, when the application is completed.

61. (New) The application execution apparatus of Claim 58,
wherein when at least two applications are completed at the same time, the
notifying means notifies each library unit which has requested notification of completion of any
of the applications, of the completion of the application, and

the collecting means specifies resources corresponding to the notified applications
based on the table, and collects the specified resources.

62. (New) The application execution apparatus of Claim 58,
wherein the requesting means further requests, if the resource providing means
provides the resource to the application, the kernel unit to notify the library unit of suspension of
the application when the application is suspended,

the notifying means notifies, when the application is suspended, the library unit
which has requested for the notification, of the suspension of the application, and

each library unit further includes

judging means for judging whether the resource provided to the application
should be collected, depending on whether the application has been completed or suspended, and

the collecting means collects the resource, when the judging means judges that the
resource should be collected.

63. (New) The application execution apparatus of Claim 62,
wherein the notifying means generates an application information instance storing
application information for identifying each application which is in execution,

the resource providing means generates, when the application being executed requests the provision of the device, a resource collection instance storing the table and a method of retrieving information from the table,

the requesting means requests the notifying means to notify the completion of the application and the suspension of the application, by registering the resource collection instance into an application information instance storing application information for identifying the application provided with the device as the resource, and

the notifying means notifies the library unit of the completion of the application when the application is completed and the suspension of the application when the application is suspended, by calling the method in the resource collection instance registered in the application information instance by the requesting means.

64. (New) An application execution apparatus including a plurality of devices, a Java middleware unit, and an OS (Operating System) unit which provides resources to applications, comprising

a storing unit for storing a plurality of applications;

wherein the Java middleware unit includes:

application executing means for reading the plurality of applications in the storing unit and executing the plurality of applications;

first table holding means for holding a first table which shows a correspondence between applications, tasks corresponding to the applications, and threads which make up each task;

first table renewing means for, when a thread is generated in response to a request from an application being executed, adding a correspondence between the generated thread, the application, and a task generated for the application to the first table to thereby renew the first table; and

notifying means for, upon receiving an instruction to complete the application, notifying the OS unit of a task corresponding the application, based on the first table, and

the OS unit includes:

task generating means for generating the task for executing the application;

thread generating means for generating threads which make up the task generated by the task generating means;

controlling means for executing the generated threads to execute program codes of the application, providing a device to the application as a resource in accordance with a request from the application, and registering a correspondence between the provided resource and the task to which the threads belong, in a second table showing a correspondence between provided resources and tasks corresponding to generated threads in the first table; and

collecting means for specifying the resource corresponding to the task notified by the notifying means based on the second table, and collecting the specified resource.

65. (New) The application execution apparatus of Claim 64,
wherein the Java middleware unit further includes:
requesting means for requesting to notify the application which is being executed,
of a change of a status of a device; and
status change notifying means for notifying the application of the change, upon
detecting the change.

66. (New) The application execution apparatus of Claim 65,
wherein the requesting means requests the status change notifying means to call a
listener which waits to be informed of the change, and
the status change notifying means makes the notification by calling the listener,
upon detecting the change.

67. (New) The application execution apparatus of Claim 66,
wherein the status change notifying means generates a special thread for calling
the listener, and calls the listener by executing the special thread,
the Java middleware unit further includes:
second table holding means for holding a table showing a correspondence
between listeners, special threads, and applications; and
table renewing means for referencing the table in the second table holding means
when the requesting means requests the status change notifying means to call the listener,
judging whether the application corresponding to the listener is shown in the table, and adding
the listener to the table in correspondence with the application if the application is shown in the
table, and

the status change notifying means does not generate the special thread if the table renewing means judges that the application is shown in the table, and generates the special thread if the table renewing means judges that the application is not shown in the table.

68. (New) The application execution apparatus of Claim 67,

wherein the special thread monitors whether information showing the occurrence of the change is held in a queue which transfers information between threads, and calls the listener upon detecting that the information is held in the queue.

69. (New) The application execution apparatus of Claim 67,

wherein the special thread is in a wait state before information showing the occurrence of the change is held in a queue which transfers information between threads, and becomes active and calls the listener when the information is held in the queue.

70. (New) The application execution apparatus of Claim 64,

wherein the Java middleware unit further includes:

resource reserve thread generating means for generating a resource reserve thread for reserving resources necessary for the Java middleware unit; and

resource reserving means for reserving the resources necessary for the Java middleware unit, by executing the resource reserve thread, and

the collecting means specifies the resources corresponding to the notified task based on the table in the controlling means, and collects the specified resources, without collecting the resources reserved by the resource reserving means.

71. (New) An application execution apparatus including a plurality of devices, a Java middleware unit, and an OS unit which provides resources to applications, comprising

- a storing unit for storing a plurality of applications,
- wherein the Java middleware unit includes:
 - application executing means for reading the plurality of applications in the storing unit and executing the plurality of applications;
 - notifying means for (a) notifying the OS unit of an application which requests a resource, in accordance with a request from the OS unit, and (b) notifying the OS unit of the application when the application is completed, and
 - the OS unit includes:
 - resource providing means for, when an application being executed requests provision of a device, providing the device to the application as a resource, requesting the notifying means to notify of the application which requests the resource, and registering a correspondence of the resource and the notified application into a table; and
 - resource collecting means for specifying, when the notifying means notifies the OS unit of the application when the application is completed, the resource corresponding to the notified application based on the table, and collecting the specified resource.

72. (New) The application execution apparatus of Claim 71,

- wherein the notifying means includes:
 - loader specifying means for specifying a class loader that loaded the application which requests the resources;

table holding means for holding a table showing a correspondence between loaded applications and class loaders which loaded the applications; and

application specifying means for specifying the application corresponding to the specified class loader, based on the table in the table holding means.

73. (New) The application execution apparatus of Claim 72,
wherein the loader specifying means specifies the class loader, by referencing a stack which stores information on a caller of a class of the application.

74. (New) The application execution apparatus of Claim 73,
wherein the OS unit further includes
assigning means for assigning an application ID to each application, and
the notifying means notifies the OS unit of the application which requests the resource by notifying the OS unit of an application ID of the application, and notifies the OS unit of the application which is completed by notifying the OS unit of the application ID of the application.

75. (New) The application execution apparatus of Claim 72,
wherein the OS unit further includes
assigning means for assigning an application ID to each application, and
the notifying means notifies the OS unit of the application which requests the resource by notifying the OS unit of an application ID of the application, and notifies the OS unit of the application which is completed by notifying the OS unit of the application ID of the application.

76. (New) The application execution apparatus of Claim 71,
wherein the OS unit further includes
assigning means for assigning an application ID to each application, and
the notifying means notifies the OS unit of the application which requests the
resource by notifying the OS unit of an application ID of the application, and notifies the OS unit
of the application which is completed by notifying the OS unit of the application ID of the
application.

77. (New) A computer-readable recording medium recording a program for use in an
application execution apparatus including: a plurality of devices; a storing unit for storing a
plurality of applications; a kernel unit for controlling execution of the plurality of applications in
the storing unit; and a plurality of library units, corresponding one-to-one to the plurality of
devices, each for managing a corresponding device by executing a device control program, the
program comprising:

a resource providing step in each library unit for, when an application being
executed requests provision of a device corresponding to the library unit, providing the device to
the application as a resource, and registering a correspondence of the resource and the
application provided with the resource into a table held in the library unit;

a requesting step in each library unit for, if the resource providing step provides
the resource to the application, requesting the kernel unit to notify the library unit of completion
of the application when the application is completed;

a collecting step in each library unit for, when the kernel unit notifies the library
unit of the completion of the application in response to the request in the requesting step,

specifying the resource corresponding to the application based on the table, and collecting the specified resource;

an application executing step in the kernel unit for reading the plurality of applications in the storing unit and executing the plurality of applications; and

a notifying step in the kernel unit for, when the application being executed is completed, notifying the library unit which has requested the notification, of the completion of the application.

78. (New) The computer-readable recording medium of Claim 77,

wherein the requesting step further requests, if the resource providing step provides the resource to the application, the kernel unit to notify the library unit of suspension of the application when the application is suspended,

the notifying step notifies, when the application is suspended, the library unit which has requested for the notification, of the suspension of the application,

the program further comprises

a judging step in each library unit for judging whether the resource provided to the application should be collected, depending on whether the application has been completed or suspended, and

the collecting step collects the resource, when the judging step judges that the resource should be collected.

79. (New) A computer-readable recording medium recording a program for use in an application execution apparatus including: a plurality of devices; a storing unit for storing a plurality of applications; a Java middleware unit, and an OS unit which provides resources to

applications, the Java middleware unit including a first table holding unit for holding a first table which shows a correspondence between applications, tasks corresponding to the applications, and threads which make up each task, the program comprising:

an application executing step in the Java middleware unit for reading the plurality of applications in the storing unit and executing the plurality of applications;

a first table renewing step in the Java middleware unit for, when a thread is generated in response to a request from an application being executed, adding a correspondence between the generated thread, the application, and a task generated for the application to the first table to thereby renew the first table;

a notifying step in the Java middleware unit for, upon receiving an instruction to complete the application, notifying the OS unit of a task corresponding the application, based on the first table;

a task generating step in the OS unit for generating the task for executing the application;

a thread generating step in the OS unit for generating threads which make up the task generated by the task generating step;

a controlling step in the OS unit for executing the generated threads to execute program codes of the application, providing a device to the application as a resource in accordance with a request from the application, and registering a correspondence between the provided resource and the task to which the threads belong, in a second table which is held in the OS unit and shows a correspondence between provided resources and tasks corresponding to generated threads in the first table; and

a collecting step in the OS unit for specifying the resource corresponding to the task notified by the notifying step based on the second table, and collecting the specified resource.

80. (New) A computer-readable recording medium recording a program for use in an application execution apparatus including: a plurality of devices; a storing unit for storing a plurality of applications; a Java middleware unit; and an OS unit which provides resources to applications, the program comprising:

an application executing step in the Java middleware unit for reading the plurality of applications in the storing unit and executing the plurality of applications;

a notifying step in the Java middleware unit for (a) notifying the OS unit of an application which requests a resource, in accordance with a request from the OS unit, and (b) notifying the OS unit of the application when the application is completed;

a resource providing step in the OS unit for, when an application being executed requests provision of a device, providing the device to the application as a resource, requesting the notifying step to notify of the application which requests the resource, and registering a correspondence of the resource and the notified application into a table held in the OS unit; and

a resource collecting step in the OS unit for specifying, when the notifying step notifies the OS unit of the application when the application is completed, the resource corresponding to the notified application based on the table, and collecting the specified resource.

81. (New) An application execution apparatus for managing a memory heap area for applications which requires garbage collection, comprising:

application storing means for storing a plurality of applications;

application executing means for reading each application in the application storing means, and executing the read application by generating a plurality of objects relating to the application and executing the plurality of objects;

divided heap area acquiring means for acquiring, when the application executing means executes the application, a divided heap area in the memory heap area, the divided heap area being unique to the application;

garbage collecting means for performing garbage collection in units of divided heap areas;

allocation controlling means for, when the application executing means generates an object relating to the application, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting means to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative;

locking means for, while the garbage collecting means is performing the garbage collection on the divided heap area, suspending the execution of only the application for which the divided heap area being garbage-collected is acquired; and

memory releasing means for releasing the divided heap area, when the application is completed.

82. (New) The application execution apparatus of Claim 81,
wherein the divided heap area acquiring means includes
table holding means for holding a table showing a correspondence between
applications being executed and divided heap areas acquired for the applications, and
the memory releasing means specifies the divided heap area corresponding to the
application based on the table in the table holding means, and releases the specified divided heap
area.

83. (New) The application execution apparatus of Claim 81,
wherein the divided heap area acquiring means includes
table holding means for holding a table showing a correspondence between
applications being executed and divided heap areas acquired for the applications, and
when the application executing means executes the application, the allocation
controlling means registers a correspondence of the application and the divided heap area
acquired for the application, into the table in the table holding means.

84. (New) An application execution apparatus for managing a memory heap area for
applications which requires garbage collection, comprising:
application storing means for storing a plurality of applications;
application executing means for reading each application in the application
storing means, generating a plurality of objects relating to the application, and executing the
plurality of objects to thereby execute the application;
system heap area allocating means for allocating the memory heap area as a
system heap area;

object area acquiring means for acquiring a system-related object area in the system heap area;

divided heap area acquiring means for acquiring a divided heap area in the system heap area when the application executing means executes the application, the divided heap area being unique to the application;

garbage collecting means for performing garbage collection in units of divided heap areas;

allocation controlling means for, when the application executing means generates an object relating to the application being executed, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting means to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative; and

memory releasing means for releasing the divided heap area when the application is completed.

85. (New) A memory heap management method used in an apparatus for managing a memory heap area for applications which requires garbage collection, the apparatus including an application storing unit for storing a plurality of applications, the memory heap management method comprising:

an application executing step for reading each application in the application storing unit, and executing the read application by generating a plurality of objects relating to the application and executing the plurality of objects;

a divided heap area acquiring step for acquiring, when the application executing step executes the application, a divided heap area in the memory heap area, the divided heap area being unique to the application;

a garbage collecting step for performing garbage collection in units of divided heap areas;

an allocation controlling step for, when the application executing step generates an object relating to the application, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting step to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative;

a locking step for, while the garbage collecting step is performing the garbage collection on the divided heap area, suspending the execution of only the application for which the divided heap area being garbage-collected is acquired; and

a memory releasing step for releasing the divided heap area, when the application is completed.

86. (New) The memory heap management method of Claim 85,

wherein the apparatus includes a table holding unit for holding a table showing a correspondence between applications being executed and divided heap areas acquired for the applications, and

the memory releasing step specifies the divided heap area corresponding to the application based on the table in the table holding unit, and releases the specified divided heap area.

87. (New) The memory heap management method of Claim 85,

wherein the apparatus includes a table holding unit for holding a table showing a correspondence between applications being executed and divided heap areas acquired for the applications, and

when the application executing step executes the application, the allocation controlling step registers a correspondence of the application and the divided heap area acquired for the application, into the table in the table holding unit.

88. (New) A memory heap management method used in an apparatus for managing a memory heap area for applications which requires garbage collection, the apparatus including an application storing unit for storing a plurality of applications, the memory heap management method comprising:

an application executing step for reading each application in the application storing unit, generating a plurality of objects relating to the application, and executing the plurality of objects to thereby execute the application;

a system heap area allocating step for allocating the memory heap area as a system heap area;

an object area acquiring step for acquiring a system-related object area in the system heap area;

a divided heap area acquiring step for acquiring a divided heap area in the system heap area when the application executing step executes the application, the divided heap area being unique to the application;

a garbage collecting step for performing garbage collection in units of divided heap areas;

an allocation controlling step for, when the application executing step generates an object relating to the application being executed, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting step to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative; and

a memory releasing step for releasing the divided heap area when the application is completed.

89. (New) A computer-readable recording medium recording a program for use in an apparatus for managing a memory heap area for applications which requires garbage collection, the apparatus including an application storing unit for storing a plurality of applications, the program comprising:

an application executing step for reading each application in the application storing unit, and executing the read application by generating a plurality of objects relating to the application and executing the plurality of objects;

a divided heap area acquiring step for acquiring, when the application executing step executes the application, a divided heap area in the memory heap area, the divided heap area being unique to the application;

a garbage collecting step for performing garbage collection in units of divided heap areas;

an allocation controlling step for, when the application executing step generates an object relating to the application, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting step to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative;

a locking step for, while the garbage collecting step is performing the garbage collection on the divided heap area, suspending the execution of only the application for which the divided heap area being garbage-collected is acquired; and

a memory releasing step for releasing the divided heap area, when the application is completed.

90. (New) The computer-readable recording medium of Claim 89,

wherein the apparatus includes a table holding unit for holding a table showing a correspondence between applications being executed and divided heap areas acquired for the applications, and

the memory releasing step specifies the divided heap area corresponding to the application based on the table in the table holding unit, and releases the specified divided heap area.

91. (New) The computer-readable recording medium of Claim 89,

wherein the apparatus includes a table holding unit for holding a table showing a correspondence between applications being executed and divided heap areas acquired for the applications, and

when the application executing step executes the application, the allocation controlling step registers a correspondence of the application and the divided heap area acquired for the application, into the table in the table holding unit.

92. (New) A computer-readable recording medium recording a program for use in an apparatus for managing a memory heap area for applications which requires garbage collection, the apparatus including an application storing unit for storing a plurality of applications, the program comprising:

an application executing step for reading each application in the application storing unit, generating a plurality of objects relating to the application, and executing the plurality of objects to thereby execute the application;

a system heap area allocating step for allocating the memory heap area as a system heap area;

an object area acquiring step for acquiring a system-related object area in the system heap area;

a divided heap area acquiring step for acquiring a divided heap area in the system heap area when the application executing step executes the application, the divided heap area being unique to the application;

a garbage collecting step for performing garbage collection in units of divided heap areas;

an allocation controlling step for, when the application executing step generates an object relating to the application being executed, judging whether the object is allocable in the divided heap area acquired for the application, allocating the object in the divided heap area if the judgment is affirmative, and instructing the garbage collecting step to perform garbage collection on the divided heap area and allocating the object in the divided heap area after the garbage collection if the judgment is negative; and

a memory releasing step for releasing the divided heap area when the application is completed.